WHAT IS CLAIMED IS:

1. A method of transmitting executable software from a server to a client computer, the method comprising:

    segmenting each of a plurality of applications into a collection of executable blocks;

    forming an InitBlock Bundle comprising blocks executable during initialization of the plurality of applications, at least one block from each application being included in the InitBlock Bundle;

    sending the InitBlock Bundle to a client computer; and

    sending other blocks from the plurality of collections of executable blocks to the client computer subsequent to a start of execution of the InitBlock Bundle.

2. The method of claim 1 wherein:

    the plurality of applications comprise at least one application subscribed to by a user and at least one application not subscribed to by the user; and

    the method further comprises:

    monitoring execution of applications subscribed to by the user to determine an application usage pattern; and

    based on the usage pattern, sending data to the client terminal to display information about a first one of the unsubscribed applications.

3. The method of claim 2 wherein:

the data to display information about one of the unsubscribed applications comprises an

offer to subscribe to the first one of the unsubscribed applications.

4. The method of claim 3 further comprising:

in response to the offer to subscribe, receiving data at the server indicating acceptance of

the offer; and

sending data to the client terminal to enable execution of the first one of the unsubscribed

applications by the user.

5. The method of claim 4 wherein:

the InitBlock Bundle comprises access control data; and

sending data to enable execution comprises sending changed access control data from the

server to the client computer.

6. The method of claim 5 wherein sending the changed access control data comprises

automatically sending in response to a subscription request received at the server from

the client computer.

7. The method of claim 5 wherein the access control data comprises an encryption key

enabling access to blocks of subscribed-to applications.

8. The method of claim 1 further comprising:

    from each of a plurality of service providers, sending to a client terminal an InitBlock

        Bundle comprising a plurality of initialization blocks;

    monitoring execution of blocks in each of said InitBlock Bundles to determine a usage

        pattern; and

    forming a new InitBlock Bundle based on the usage pattern.

9. The method of claim 8 wherein the new InitBlock Bundle comprises executable blocks

    associated with applications from different ones of the service providers.

10. The method of claim 1 wherein at least one of the blocks in the InitBlock Bundle is a

    shared block executable during the initialization phase of different ones of the

    applications.

11. The method of claim 1 wherein the InitBlock Bundle comprises a set of blocks sufficient

    to enable execution of each of the plurality of applications to a point when the application

    awaits user input.

12. The method of claim 1 wherein forming the InitBlock Bundle comprises:

    monitoring usage of a plurality of different applications; and wherein

    forming the initialization block comprises forming based on the monitored usage.

13. The method of claim 1 further comprising:

    sending from the server to the client a plurality of key values, the key values identifying

        ones of the collection of collections of executable blocks;

    receiving a response at the server from the client indicating blocks identified by the key

5   values that are already stored at the client; and wherein

    sending the InitBlock Bundle comprises omitting blocks stored already stored at the

        client.

14. A method of streaming data from a server to a client computer, the method comprising:

    sending from a server to a client a key value identifying a streamable block;

10   receiving a response at the server from the client indicating whether the client has a

        locally stored copy of the block; and

    sending the block to the client if the client does not have a locally stored copy.

15. The method of claim 14 wherein:

    sending a key value further comprises sending a group of other key values identifying

15   other streamable blocks;

    receiving a response further comprises receiving a response indicating whether the client

        has locally stored copies of ones of the other blocks; and

    sending the block further comprises sending ones of the other blocks that are not locally

        stored at the client.

16. The method of claim 15 further comprising:

at the client, storing first data associating key values with locally stored blocks; and

processing the first data to determine whether the client has a locally stored copy of a

block identified by the received key value.

5   17. The method of claim 14 wherein the key value is computed at the server using a hashing

algorithm.

18. The method of claim 17 wherein the hashing algorithm comprises a digital signature

algorithm.

19. A computer system comprising:

10   a database storing a plurality of executable applications segmented into a plurality of

code blocks, each application's plurality of code blocks comprising a set of

initialization code blocks;

a processor operatively coupled to a network interface, to the database and to a computer

readable data storage media comprising instructions to configure the processor to:

15   form an initialization block comprising initialization code blocks for at least two of

the plurality of applications; and

send the initialization block to a client computer operatively coupled to the network

interface.

20. The system of claim 19 wherein the data storage media further comprises instructions to configure the processor to:

    monitoring execution of initialization code blocks at the client computer to determine a usage pattern; and

5   forming a new InitBlock Bundle based on the usage pattern.

21. The system of claim 19 wherein:

    the system further comprises a database comprising a plurality of user profiles, each user profile comprising security data to control usage of ones of the plurality of applications by a respective user;

10   the data storage media further comprises instructions to query the database of user profiles to access security data associated with a first user;

    instructions to process the security data to determine application restriction data associated with the first user; and

    instructions to send the application restriction data to the first client computer.

15 22. The system of claim 21 wherein the application restriction data comprises further comprises data preventing user access to the unsubscribed second application.

23. A computer readable data storage apparatus storing instructions for configuring a

computer to:

send to a client terminal a key value identifying a streamable block;

receive a response from the client terminal indicating whether the client terminal has a

5   locally stored copy of the block; and

send the block to the client terminal if the client does not have a locally stored copy.

24. The apparatus of claim 23 wherein:

the instructions to send a key value further comprises instructions to simultaneously send

a group of other key values identifying other streamable blocks;

10   the instructions to receive a response further comprises instructions to receive a response

indicating whether the client terminal has locally stored copies of ones of the other

blocks; and

the instructions to send the block further comprises instructions to send ones of the other

blocks that are not locally stored at the client.